

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1. (Currently amended) A computer-implemented diagram system for manipulating a domain specific language instance model via a diagram, comprising:
 - a diagram that stores at least one shape element in accordance with object role modeling;
 - an object model application program interface that includes a control that facilitates access to the diagram, the control maintaining state information associated with the diagram and with an underlying data document, employing a common framework to keep both the diagram and underlying data synchronized; and
 - a computer readable storage medium comprising sets of code and data structures for causing a computer to modify the diagram with the object model application program interface, wherein underlying data document is a domain specific language model representing both model elements and presentation elements.
2. (Previously presented) The computer-implemented system of claim 1, the state information comprising at least one of selection, zoom and scroll position.
3. (Previously presented) The computer-implemented system of claim 1, the control captures operating system events.
4. (Previously presented) The computer-implemented system of claim 3, the control providing at least some of the operating system events to the shape element.
5. (Previously presented) The computer-implemented system of claim 1, the control rerouting at least one of paint, keyboard and mouse events to at least one of the diagram and the shape element.

6. (Previously presented) The computer-implemented system of claim 1, the diagram and the shape element responsible for painting themselves.
7. (Previously presented) The computer-implemented system of claim 1, the diagram or the shape element being responsible for responding to a user interaction.
8. (Previously presented) The computer-implemented system of claim 1, the shape element being based, at least in part, upon a model element class.
9. (Previously presented) The computer-implemented system of claim 1, the diagram being based, at least in part, upon a node shape that has a bounds property which defines its location and size, the node shape derived from the shape element.
10. (Previously presented) The computer-implemented system of claim 1, at least one shape element having a child shape element.
11. (Previously presented) The computer-implemented system of claim 1, the shape element derived from a presentation element.
12. (Previously presented) The computer-implemented system of claim 1, the shape element comprising at least one of a geometry property, a style set property and a shape fields property.
13. (Previously presented) The computer-implemented system of claim 1, the diagram having a graph object employed for hittesting for testing a shape that has been user dropped by dragging.
14. (Previously presented) The computer-implemented system of claim 1, the shape element being control-less.
15. (Currently amended) A computer-implemented method that facilitates access to a diagram comprising:

employing a control to access a diagram ; and,
storing at least one shape element contained by the diagram,
wherein each shape is directly linked to a domain specific language elements that the shape represents thereby facilitating manipulation of a domain specific language instance model via the diagram.

16. (Previously presented) The computer-implemented method of claim 15, the control maintaining state information associated with the diagram.

17. (Previously presented) A computer readable medium having stored thereon computer executable instructions for carrying out the method of claim 15.

18. (Cancelled).

19. (Currently amended) A computer readable medium storing computer executable components of a diagram system comprising:

a diagram component that stores at least one shape element and has a graph object employed for hittesting for testing a shape that has been user dropped by dragging; and,
an application program interface component that includes a control that facilitates access to the diagram, the control maintaining state information associated with the diagram.

20. (Currently amended) A computer-implemented diagram system comprising:
means for storing at least one shape element in a diagram, the diagram and/or the shape element being responsible for painting themselves and responding to a user interaction;
means for accessing the diagram; and,
means for maintaining state information associated with the diagram,
wherein at least one shape element having a child shape element.

21. (Currently amended) A method for diagramming, comprising:

managing presentation elements comprised of diagrams and shapes in a same context as correspondingly depicted design elements of a diagram on design surface in an object model diagramming system to avoid synchronization issues of mirrored presentation and design classes;

providing an object model application programming interface comprising a single diagram control for the design surface that maintains state information associated with the diagram by capturing events; and

rendering shapes of the diagram that are responsible for painting themselves and for responding to user interaction via a user interface, making implementation very light weight and independent of any specific diagram.